## **CLAIMS**

What is claimed is:

- 1. A method, comprising:
- 2 inserting an allocation instruction within a routine if a function call instruction is
- 3 found within said routine.
- 1 2. The method oxclaim 1 further comprising configuring said allocation
- 2 instruction to allocate only for the live information that exists within said routine
- 3 when said inserted allocation instruction is executed.
- 1 3. The method of claim 2 wherein said live information is determined by
- 2 identifying information that is referred to before and after said function call.
- 1 4. The method of claim 3 wherein said information identified after said
- 2 function call extends to an exit block of said routine.
- 1 5. The method of claim 4 wherein the worst case path to said exit block is
- 2 allocated for.
- 1 6. The method of claim 3 wherein said information identified after said
- 2 function call extends to a post-dominator block of said routine.
- 7. The method of claim 6 wherein the worst case path to said post-dominator
- 2 block is allocated for.



- The method of claim 2 wherein said live information is information that is
- 2 local to said routine.
- 1 9. The method of claim 8 wherein the processor said routine is to be
- 2 executed upon has its associated register space partitioned into register space
- 3 used only for local information and register space used only for global
- 4 information, said allocation instruction pertaining only to said register space
- 5 used for local information.
- 1 10. The method of claim 2 wherein said live information includes global
- 2 information.
- 1 11. The method of claim 1 wherein said allocation instruction is inserted just
- 2 before said function call.
- 1 12. The method of claim 1 wherein said allocation instruction is inserted in a
- 2 pre-dominator basic block of said function call.
- 1 13. The method of claim 12 wherein said allocation instruction is inserted in
- 2 said pre-dominator basic block of said function call if there exists a post-
- 3 dominator basic block of said function call.
- 1 14. A method comprising:

inserting multiple allocation instructions within a routine by searching for one or

- 3 more functional characteristics within said routine and inserting an allocation
- 4 instruction within said routine if a functional characteristic is discovered.
- 1 15. The method of claim 14 wherein a loop in a control flow graph
- 2 corresponds to a functional characteristic.
- 1 16. The method of claim 15 wherein said allocation instruction inserted for
- 2 said loop is inserted above said loop.
- 1 17. The method of claim 16 wherein said allocation instruction allocates for a
- 2 worst case path to an exit block of said routine.
- 1 18. The method of claim 16 wherein said allocation instruction allocates for a
- worst case path to a post-dominator block of said routine.
- 1 19. The method of claim 14 wherein a software pipelined loop corresponds to
- 2 a functional characteristic.
- 1 20. The method of claim 19 wherein said allocation instruction inserted for
- 2 said software pipelined loop is inserted above said loop
- 1 21. The method of claim 20 wherein said allocation instruction allocates for a
- 2 worst case path to an exit block of said routine.
- 1 22. The method of claim 21 wherein said allocation instruction allocates for a
- worst case path to a post-dominator block of said routine.



- 23. The method of claim 14 wherein a function call corresponds to a
- 2 functional characteristic.
- 1 24. The method of claim 14 further comprising determining the number of
- 2 registers to be allocated for an allocation instruction after a functional
- 3 characteristic is found.
- 1 25. The method of daim 24 wherein all functional characteristics within said
- 2 routine are discovered before said determining is performed.
- 1 26. The method of claim 24 wherein said determining is performed before a
- 2 next functional characteristic is discovered.
- 1 27. The method of claim 14 further comprising building an understanding of
- 2 said routine's control flow graph before said searching is performed.
- 1 28. A method, comprising:
- a) performing a first allocation for a tirst amount of register space at
- 3 the entry block of a routine;
- 4 b) performing a second allocation for a second amount of register
- 5 space for the live information within said routine at the time of said second
- 6 allocation;
- 7 c) performing a function call to a second routine;

- d) performing a third allocation for a third amount of register space at
- 9 the entry block of said second routine, said third amount of register space and
- 10 said first amount register space having a common register.
  - 1 29. The method of claim 28 wherein said live information is determined by
  - 2 identifying information that is referred to before and after said function call.
  - 1 30. The method of claim 29 wherein said information identified after said
  - 2 function call extends to an eximplock of said routine.
  - 1 31. The method of claim 30 wherein the worst case path to said exit block is
  - 2 allocated for.
  - 1 32. The method of claim 29 wherein said information identified after said
  - 2 function call extends to a post-dominator block of said routine.
  - 1 33. The method of claim 32 wherein the worst case path to said post-
  - 2 dominator block is allocated for.
  - 1 34. The method of claim 28 wherein said live information is information that
  - 2 is local to said routine.
  - 1 35. The method of claim 34 wherein the processor said toutine is to be
  - 2 executed upon has its associated register space partitioned in pregister space
  - 3 used only for local information and register space used only for global

information, said allocation instruction pertaining only to said register space used for local information.

- 1 36. The method of claim 28 wherein said live information includes global
- 2 information
- 1 37. The method of claim 28 wherein said second allocation is performed just
- 2 before said function call.
- 1 38. The method of claim 28 wherein said second allocation is performed in a
- 2 pre-dominator basic block of said function call.
- 1 39. The method of claim 28 wherein said second allocation is performed in
- 2 said pre-dominator basic block of said function call if there exists a post-
- 3 dominator basic block of said function call.
- 1 40. The method of claim 28 further comprising compiling said routine.
- 1 41. A machine readable medium having stored thereon sequences of
- 2 instructions which are executable by a digital processing system, and which,
- 3 when executed by the digital processing system, cause the system to perform a
- 4 method comprising:
- 5 inserting an allocation instruction within a routine if a function call
- 6 instruction is found within said routine.

42. The machine readable medium of claim 41 further comprising sequences

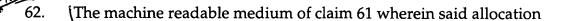
- 2 of instructions which cause the system to configure said allocation instruction to
- 3 allocate only for the live information that exists within said routine when said
- 4 inserted allocation instruction is executed.
- 1 43. The machine readable medium of claim 42 wherein said live information
- 2 is determined by identifying information that is referred to before and after said
- 3 function call.
- 1 44. The machine readable medium of claim 43 wherein said information
- 2 identified after said function call extends to an exit block of said routine.
- 1 45. The machine readable medium of claim 44 wherein the worst case path to
- 2 said exit block is allocated for.
- 1 46. The machine readable medium of claim 43 wherein said information
- 2 identified after said function call extends to a post-dominator block of said
- 3 routine.
- 1 47. The machine readable medium of claim 46 wherein the worst case path to
- 2 said post-dominator block is allocated for.
- 1 48. The machine readable medium of claim 42 wherein said live information
- 2 is information that is local to said routine.

49. The machine readable medium of claim 48 wherein the processor said

- 2 routine is to be executed upon has its associated register space partitioned into
- 3 register space used only for local information and register space used only for
- 4 global information, said allocation instruction pertaining only to said register
- 5 space used for local information.
- 1 50. The machine readable medium of claim 42 wherein said live information
- 2 includes global information.
- 1 51. The machine readable medium of claim 41 wherein said allocation
- 2 instruction is inserted just before said function call.
- 1 52. The machine readable medium of claim 41 wherein said allocation
- 2 instruction is inserted in a pre-dominator basic block of said function call.
- 1 53. The machine readable medium of claim 52 wherein said allocation
- 2 instruction is inserted in said pre-dominator basic block of said function call if
- 3 there exists a post-dominator basic block of said function call.
- 1 54. A machine readable medium having stored thereon sequences of
- 2 instructions which are executable by a digital processing system, and which,
- 3 when executed by the digital processing system, cause the system to perform a
- 4 method comprising:
- 5 inserting multiple allocation instructions within a routine by searching for
- 6 one or more functional characteristics within said routine and inserting an

allocation instruction within said routine if a functional characteristic is

- 8 discovered.
- 1 55. The machine readable medium of claim 54 wherein a loop in a control
- 2 flow graph corresponds to a functional characteristic.
- 1 56. The machine readable medium of claim 55 wherein said allocation
- 2 instruction inserted for said loop is inserted above said loop.
- 1 57. The machine readable medium of claim 56 wherein said allocation
- 2 instruction allocates for a worst case path to an exit block of said routine.
- 1 58. The machine readable medium of claim 56 wherein said allocation
- 2 instruction allocates for a worst case path to a post-dominator block of said
- 3 routine.
- 1 59. The machine readable medium of claim 54 wherein a software pipelined
- 2 loop corresponds to a functional characteristic.
- 1 60. The machine readable medium of claim 59 wherein said allocation
- 2 instruction inserted for said software pipelined loop is inserted above said loop.
- 1 61. The machine readable medium of claim 60 wherein said allocation
- 2 instruction allocates for a worst case path to an exit block of said routine.



- 2 instruction allocates for a worst case path to a post-dominator block of said
- 3 routine.
- 1 63. The machine readable medium of claim 54 wherein a function call
- 2 corresponds to a functional characteristic.
- 1 64. The machine readable medium of claim 54 further comprising sequences
- 2 of instructions which cause the system to determine the number of registers to be
- 3 allocated for an allocation in truction after a functional characteristic is found.
- 1 65. The machine readable medium of claim 64 wherein all functional
- 2 characteristics within said routine are discovered before said determining is
- 3 performed.
- 1 66. The machine readable medium of claim 64 wherein said determining is
- 2 performed before a next functional characteristic is discovered.
- 1 67. The machine readable medium of claim 54 further comprising sequences
- 2 of instructions which cause the system to build an understanding of said
- 3 routine's control flow graph before said searching is performed.